

FIG. 1

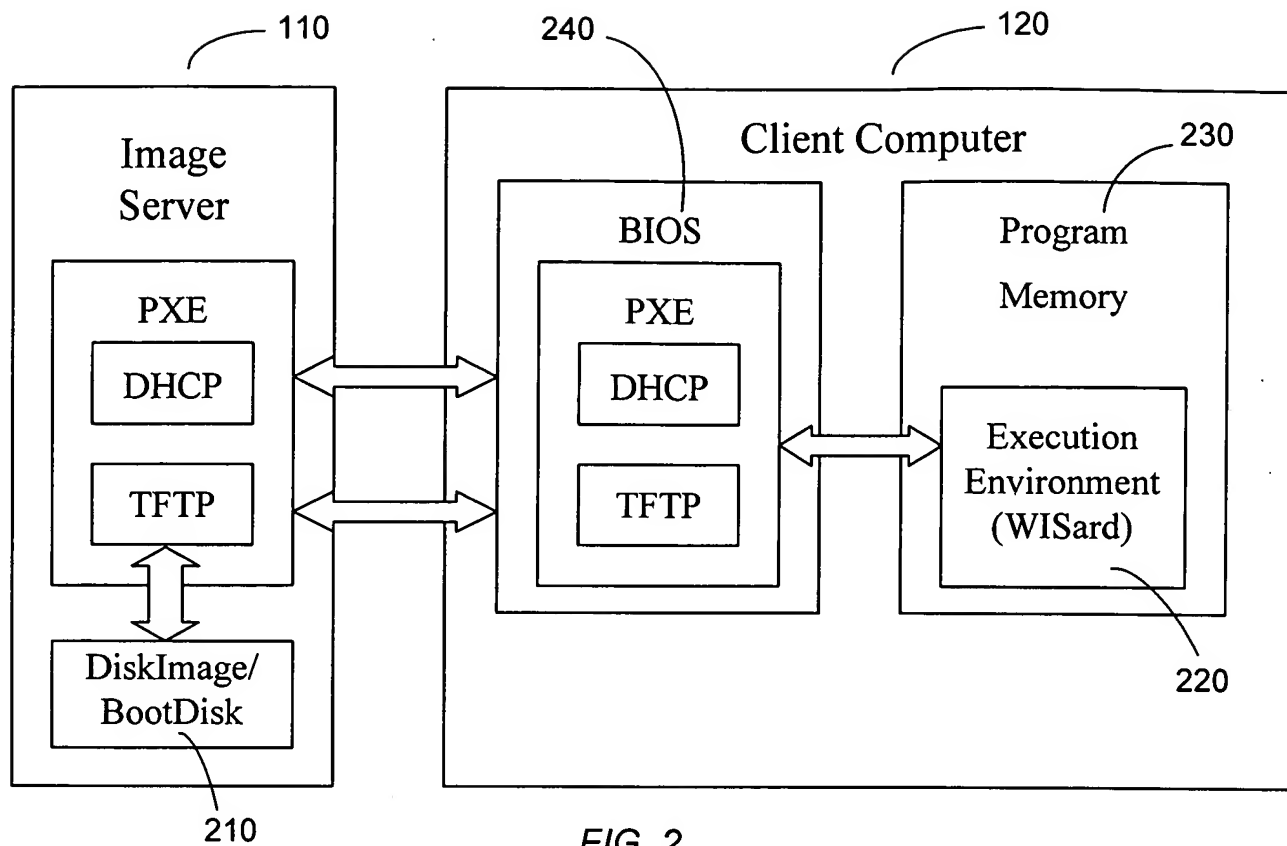


FIG. 2

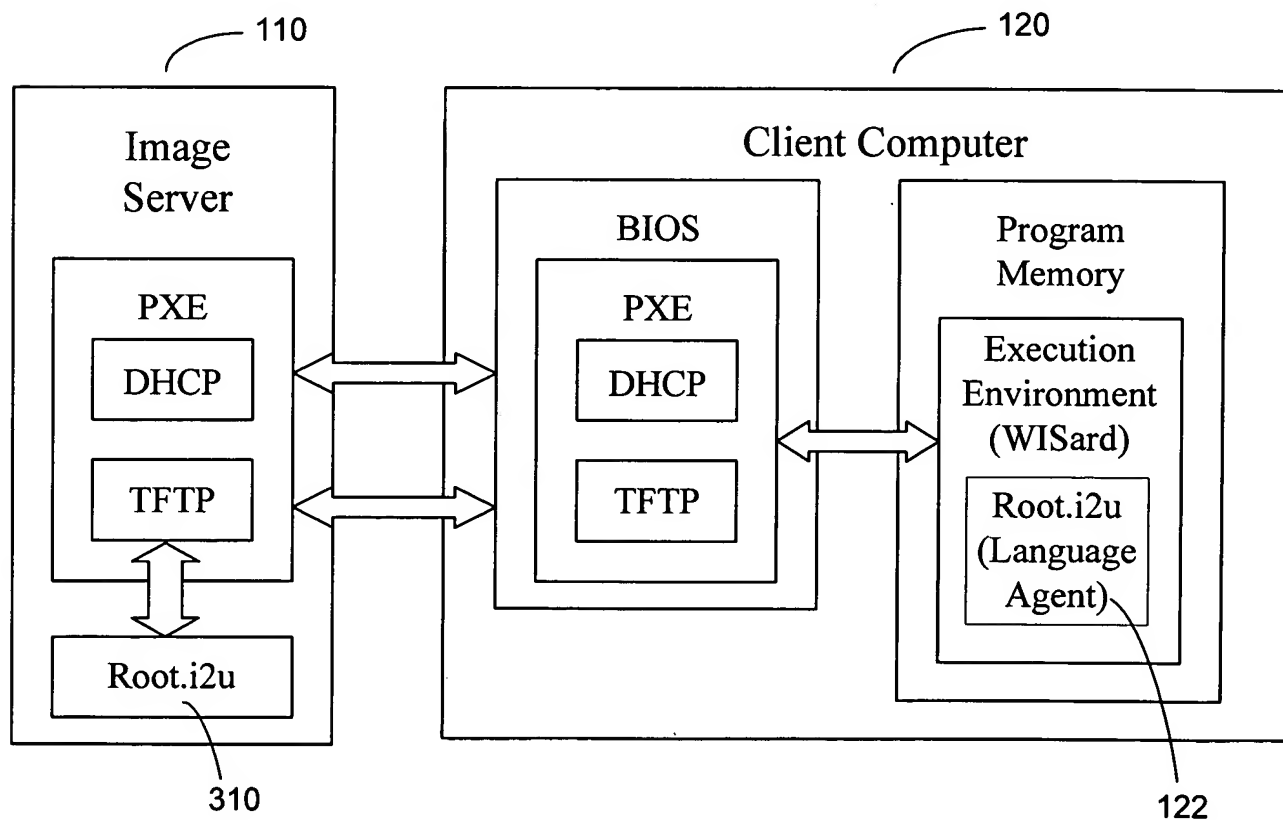


FIG. 3

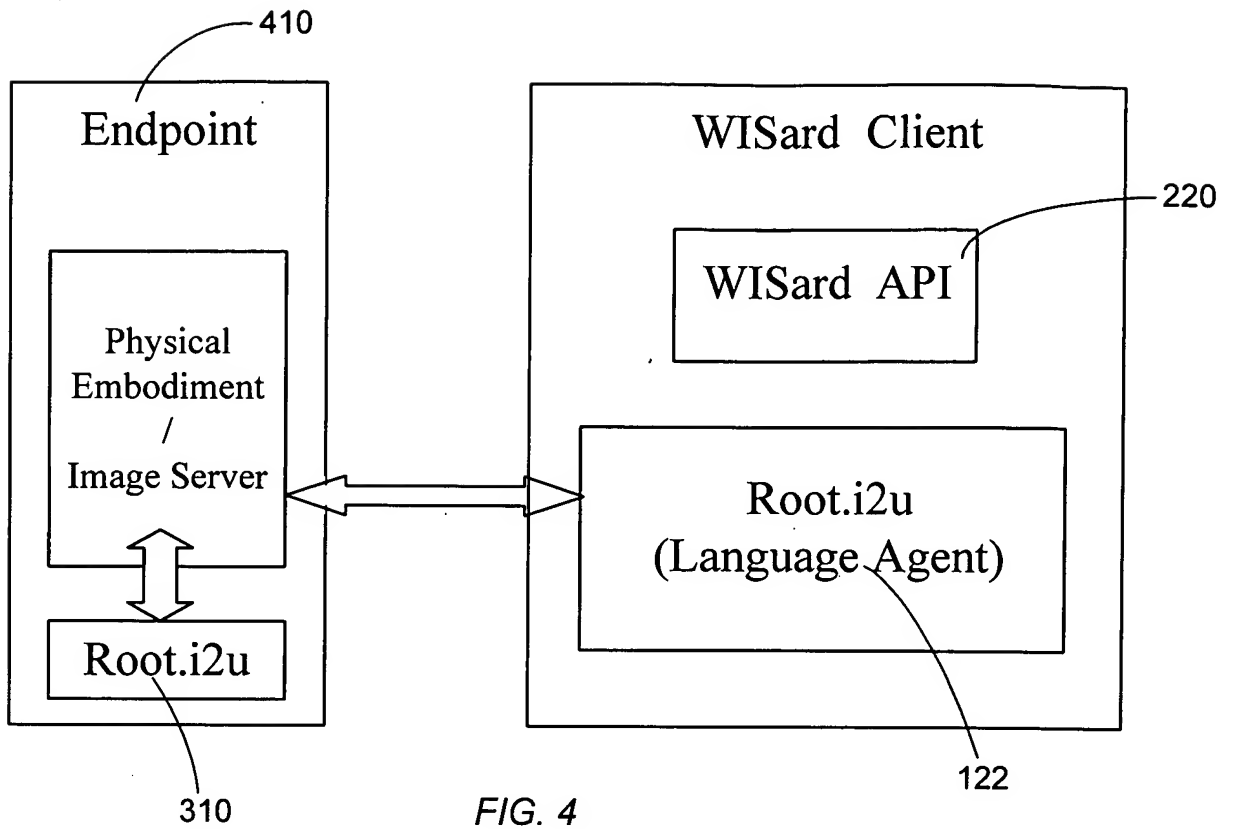


FIG. 4

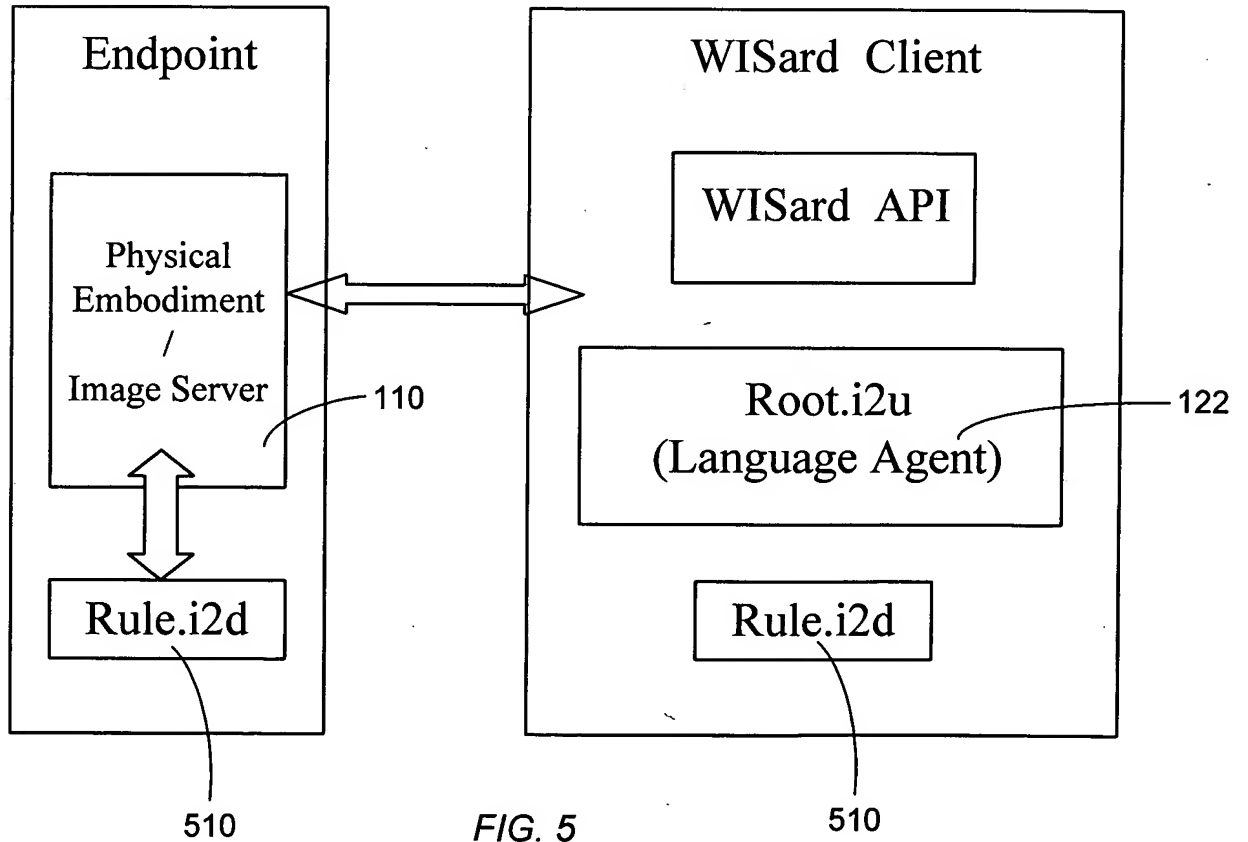


FIG. 5

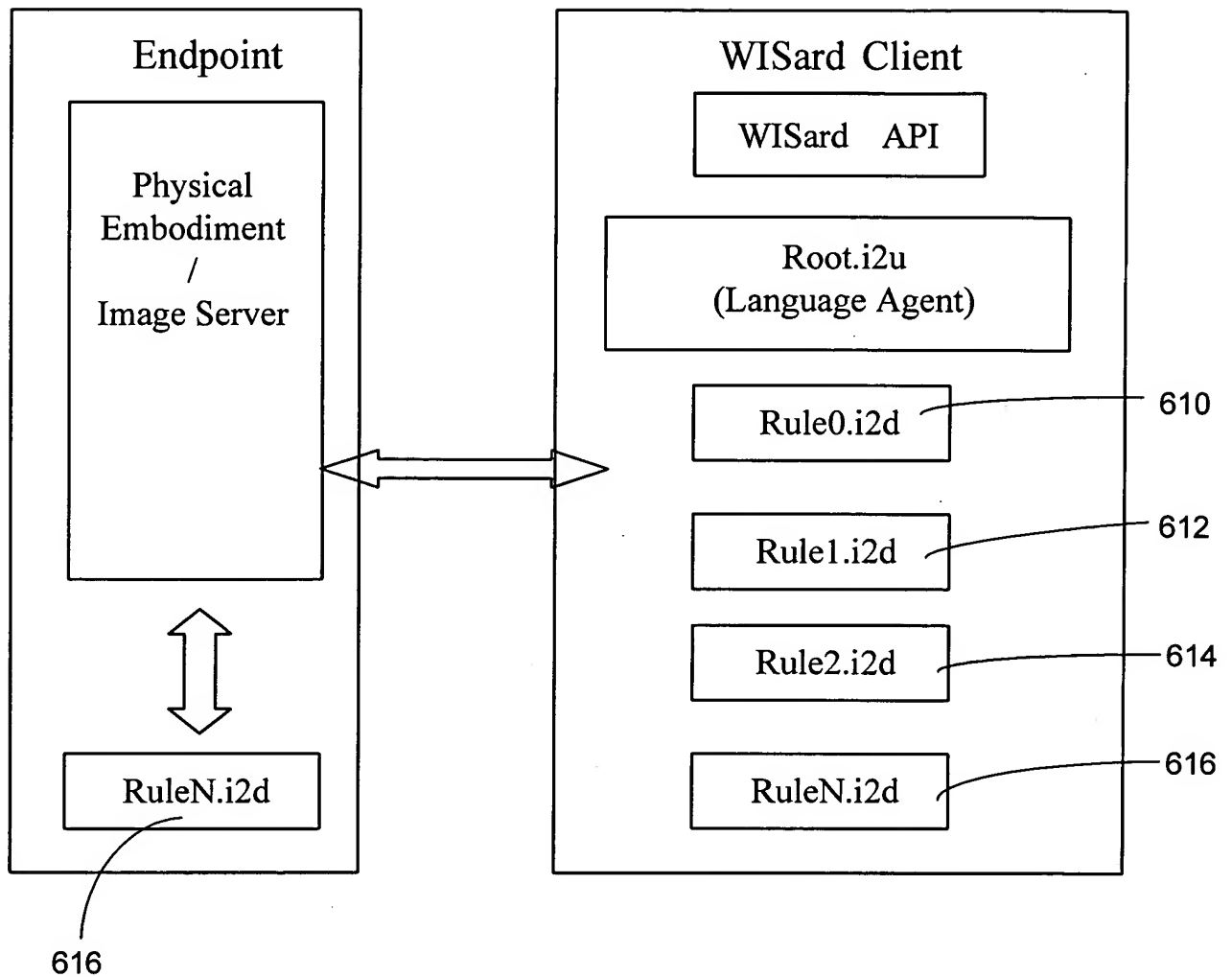


FIG. 6

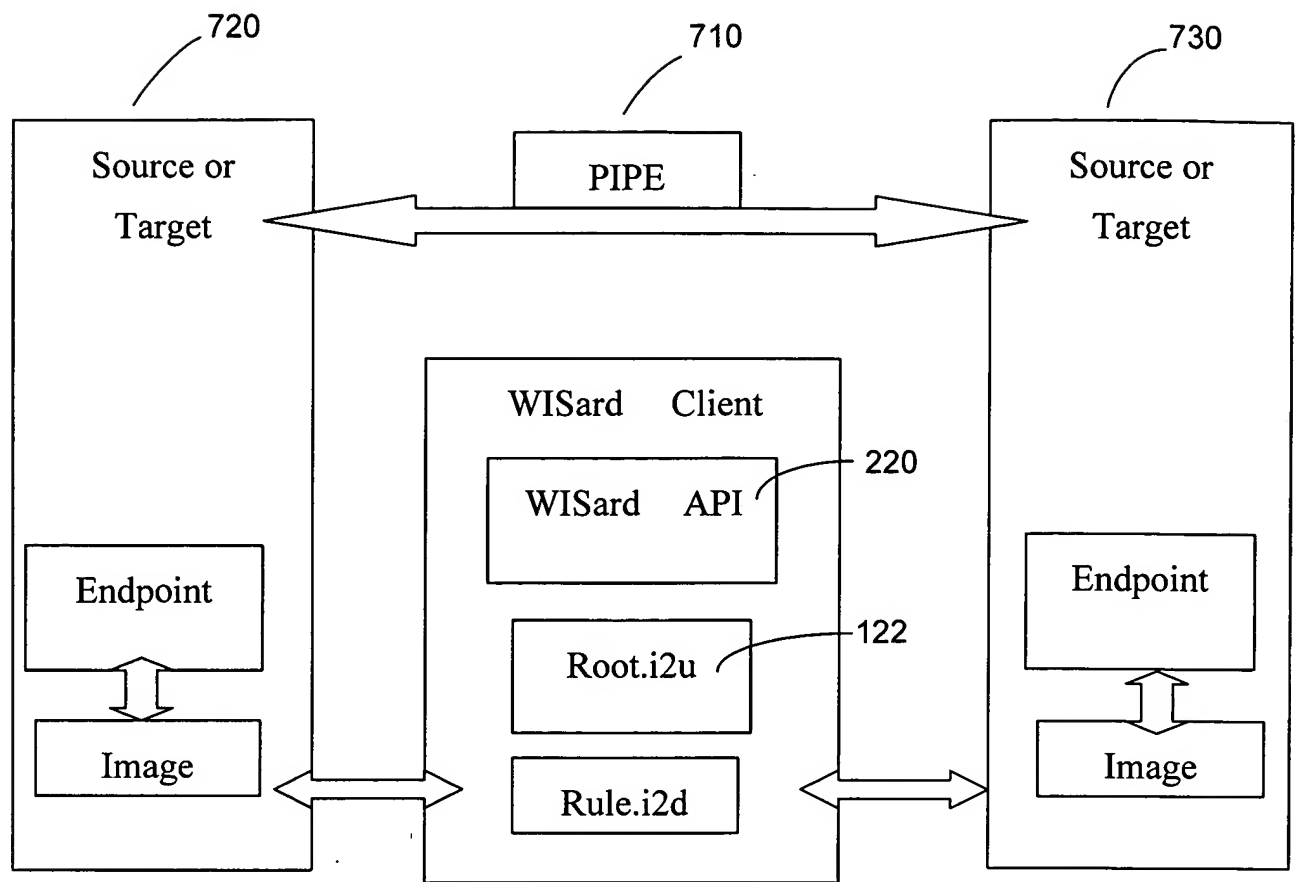


FIG. 7

```

@echo off
REM *****
REM Get functional script for reading from generic "gen" devices.
REM
REM (c) 2003 Wyse Technology
REM
REM Script "geti2d.bat" Begin
REM
REM Output: "get.i2d" Partition Type 2
REM
REM set gVERSION = "get.i2d Version 8.5"
REM
REM *****
REM ***** Remove OLD archive
del get.i2d

REM *****
REM Display GET version
set gVERSION > ver.msg
file2img ver.msg
del ver.msg
REM *****
REM ***** Check for OVERLOAD
file2img ~if archive/get.i2d
file2img "-<SRCLproto><SRCdevice>getOVL.i2d" archive/get.i2d
file2img "-archive/<SRCTMPproto><SRCTMPdevice>get.i2d" archive/get.i2d
file2img ~else archive/get.i2d

REM *****
REM Now we create an archive that creates an image for a drive
REM *****
file2img ~encode archive/get.i2d "-archive/<ENCproto><ENCdevice><ENCimage>"
REM *****
REM At extraction time, all syntax will be placed into the runtime class "<ENCproto><ENCdevice><ENCimage>"
REM *****

REM *****
REM Script "put.i2d" Generator Begin
REM *****

REM ***** Remove OLD stuff

REM ***** Create messages
echo NAND DRIVE 00
echo No REGISTRY found during RAPPORT restoration (atf) > fake.msg

```

FIG. 8a

```

REM *****
REM *****
REM ***** Validate Security Key *****
file2img "~<SRCsec>" archive/get.i2d "<DSTsec>"

REM ***** Specify type of test/perform test *****
file2img "~<SRCTest>" archive/get.i2d "<DSTtest>"

REM *****
REM ***** Check for hostname.arc and copy if there *****
file2img ~if archive/get.i2d
file2img "~<DSTLCLproto><DSTdevice>hostname.arc" archive/get.i2d "<DSTTMPproto><DSTTMPdevice>hostname.arc"
file2img ~if archive/get.i2d
file2img "~<DSTLCLproto><DSTdevice>wfacss.reg" archive/get.i2d "<DSTTMPproto><DSTTMPdevice>wfacss.reg"
file2img ~else archive/get.i2d "<DSTTMPproto><DSTTMPdevice>wfacss.reg"
file2img ~endif archive/get.i2d "<DSTTMPproto><DSTTMPdevice>wfacss.reg"
file2img ~if archive/get.i2d
file2img "~<DSTLCLproto><DSTdevice>wfacss0.reg" archive/get.i2d "<DSTTMPproto><DSTTMPdevice>wfacss0.reg"
file2img ~else archive/get.i2d
file2img ~endif archive/get.i2d
file2img ~endif archive/get.i2d

REM *****
REM ***** Netxfer Model for Archive Creation *****
REM ***** Copy NOR FLASH *****
file2img "~<GET09>" archive/get.i2d "<PUT09>"
REM ***** Copy CMOS FLASH *****
file2img "~<GET06>" archive/get.i2d "<PUT06>"
REM ***** Signal Event Format *****
file2img "~<GET01>" archive/get.i2d "<PUT01>"
REM ***** Copy Allocation/MBR Data *****
file2img "~<GET07>" archive/get.i2d "<PUT07>"
REM ***** Partition FIX *****
file2img "~<GET03>" archive/get.i2d "<PUT03>"
REM ***** Copy Partition IMAGE *****
file2img "~<GET08>" archive/get.i2d "<PUT08>"
file2img "~<GET03>" archive/get.i2d "<PUT03>"
REM ***** Partition FIX *****
file2img "~<GET04>" archive/get.i2d "<PUT04>"
REM ***** Copy VERSIONS *****
file2img "~<GET05>" archive/get.i2d "<PUT05>"

REM *****
REM ***** RESTORE hostname.arc if there *****
file2img ~if archive/get.i2d
file2img "~<DSTTMPproto><DSTTMPdevice>hostname.arc" archive/get.i2d "<DSTLCLproto><DSTdevice>hostname.arc"
file2img ~if archive/get.i2d
file2img "~<DSTTMPproto><DSTTMPdevice>wfacss.reg" archive/get.i2d "<DSTLCLproto><DSTdevice>wfacss.reg"
file2img ~endif archive/get.i2d
file2img ~if archive/get.i2d
file2img "~<DSTTMPproto><DSTTMPdevice>wfacss0.reg" archive/get.i2d "<DSTLCLproto><DSTdevice>wfacss0.reg"
file2img ~endif archive/get.i2d
file2img ~endif archive/get.i2d

```

FIG. 8b

```

REM *****
REM ***** Mark End of Transfer (give control back to calling script)
file2img ~EOT      archive/get.i2d      ~EOT
REM *****

REM ***** Cleanup
del formatmsg.msg
del fake.msg
set gVERSION=

REM *****
REM Script "putgeni2d.bat" End
REM *****

REM *****
file2img ~encodeoff      archive/get.i2d
REM *****

REM ***** Undo Class-overload test
file2img ~endif      archive/get.i2d

REM ***** Mark End of Transfer (give control back to calling script)
file2img ~EOT      archive/get.i2d      ~EOT
REM *****

REM *****
REM Script "getgeni2d.bat" Generator End
REM *****

```

FIG. 8c

[illegible]